Why Binary Merge?

Koji Arikawa

Momoyama Gakuin University (St. Andrew's University, Osaka, Japan)

karikawa@andrew.ac.jp

Linguists have accumulated evidence for binary merge. This paper adds mathematical reason. The computational procedures of human natural language (C_{HL}) choose binary merge because it can contain more information, and lacks optimal balance. Nature organizes network currents to minimize information loss. This is the physical principle of minimal computation (MC). C_{HL} obeys MC. Nature has created C_{HL} in the human brain. Nature has selected binary merge whose balance is not optimal, and information loss (error) is not minimized. However, this unsettled unbalance (asymmetry) of binary merge drives the infinite growth of binary trees.

Keywords: *: balance, C_{HL}, graph, linear algebra, MC, merge*

***Journal of Cognitive Science* 17-4: 499-546, 2016**
Date submitted: 09/08/16 Date reviewed: 11/30/16
Date confirmed for publication: 12/14/16

1. Introduction

Linguists have accumulated evidence for binary merge. The binary versus ternary merge originates in the debate on configurationality, i.e., flat versus hierarchical. Consider subject (S), object (O), and verb (V). Imagine a mobile. Put aside issues of ordering, merged node terminologies, and additional head insertion.

(1) *Binary merge* versus *ternary merge*: Which one is correct?

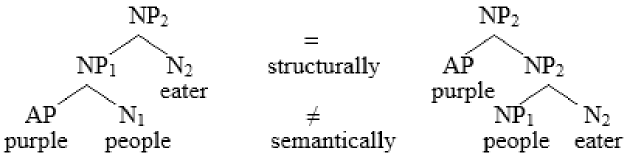
Binary merge (hierarchical) *Ternary merge (flat)*

Two hypotheses exist. Hypothesis 1: merge is parameterized as binary or ternary. For example, English and Chinese are set as binary, while Warlpiri and Japanese are fixed as ternary (e.g., Hale 1982). The former shows rigid word order, whereas the latter exhibits free word order (scrambling). Hypothesis 2: merge is binary (e.g., Saito and Hoji 1983). Let us consider a simpler example.

(2) purple people eater
  a. 'eater of purple people,' OR
  b. 'people eater that is purple,' but NOT
  c. 'both the people and eater are purple'

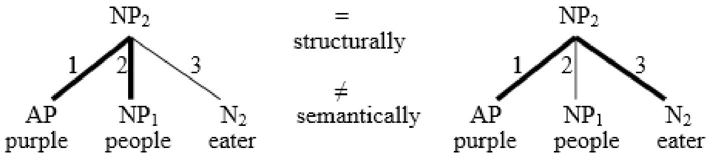
How can we differentiate (2a) and (2b) while avoiding the "mixed" reading (2c)? Structurally, the computational procedures of human natural language (C_{HL}) obeys the associative law, $[_{NP2}[_{NP1}$ *purple* + *people*] + *eater*]] = $[_{NP2}$ *purple* + $[_{NP2}$ *people* + *eater*]]. Both outputs are the same NP_2. This resembles addition, e.g., (1 + 2) + 3 = 1 + (2 + 3). On the other hand, C_{HL} does not obey the associative law semantically, i.e., (*purple* + *people*) + *eater* ≠ *purple* + (*people* + *eater*); (2a) and (2b) are different. This is similar to addition with multiplication: 1 • 2 + 3 ≠ 1 + 2 • 3 (Honma 1975: 241). Thus, *purple* • *people* + *eater* ≠ *purple* + *people* • *eater*. The asymmetry is accounted for if the modifier and the modified term are in a mutual c-command relation (Miyagawa 1989). The binary merge explains the (dis) obedience to the associative law. The subscripts indicate head projection.

(3) Binary *merge* and the (dis)obedience to the associative law



How does the ternary merge account for structure and meaning? Here, we number the edges and bold lines indicate activated edges.

(4) Ternary *merge* and the different edge activations



Assume that a phrase is mono-headed. These ternary merge forms are structurally identical, i.e., they are both NP_2. Given a mutual c-command, ternary merge incorrectly predicts the mixed reading because the adjective mutually c-commands both *people* and *eater*. To adopt ternary merge, edges 1 and 2 must activate to yield the "purple people" reading, while edges 1 and 3 activate to produce the "purple eater" reading. The activation does not occur simultaneously (no mixed reading). What causes this difference in activation? The absence of mixed reading indicates that C_{HL} does not obey the distributive law in terms of semantic interpretation, i.e., *purple* (*people* + *eater*) ≠ *purple people* + *purple eater*. However, the ternary merge with mutual c-command incorrectly predicts that C_{HL} should obey the distributive law. How does merge differ from addition and multiplication? How is the binary merge versus ternary merge issue relevant to arithmetic laws? Puzzles aside, the mutual c-command condition seems to favor the binary merge. Can we find other reasons for adopting binary merge? Is there any other reason that binary merge is "better"?

In this article, we investigate binary versus ternary merge mathematically, i.e., through linear algebra and graph theory. We calculate balance that is hidden in the original structure. How does graph theory illuminate the merge problem? How does graph theory distinguish binary merge and ternary merge? How do they differ mathematically? We employ graph theory, which is an application of linear algebra, and calculate equilibrium (balance) force hidden in binary and ternary merge.[1] The remainder of this paper is organized as follows. Section 2 lists other possible structures (trees and graphs) for the expression "purple people eater." Section 3 introduces a toy model of the graph theory using a complete graph to demonstrate how balance is hidden in a structure. Graph theory can be applied to explain how nature balances networks and distributes currents to minimize heat loss. Section 4 applies graph theory to the binary versus ternary merge problem for C_{HL}. In Section 5, we compare the hidden balance of various networks including those calculated in the appendix. Conclusions are given in Section 6. In the appendix, the balance of other possible trees and graphs is calculated.

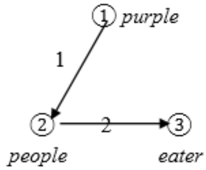
---

[1] One reviewer insightfully refers to a debate between set-theoretic and graph-theoretic approaches to phrase structure (PS), and asks why we chose to rely on graph theory. The reviewer provides us with the following excellent overview of the fundamental issue. Lasnik (2000: 29-33, 57-58), exemplifying the PS tree partially as a unary (non-branching) projection, points out that Chomsky's (1957) set-theoretic formalization of PS was based on an unordered "*is a*" relation, and was more minimalistic than Kayne's (1984) "graph-theoretic representations that encode some information (order: what projects to what) beyond the '*is a*' relation" (Lasnik 2000: 33). However, if unary projections are excluded for independent reasons, it is not clear which of the two approaches is more minimalistic. In fact, unary projections (string trees) are too linearly independent (minimalistic) from the beginning and there is no room for cost reduction. C_{HL} is not a perfectionist throughout. See appendix. According to Stewart (1995: 61), the coordinate geometry with an ordered set G, where $(\alpha, \beta) \neq (\beta, \alpha)$, is a subset of an unordered set S, where $\{\alpha, \beta\} = \{\beta, \alpha\}$. Thus, $S \supseteq G$. In other words, S contains G (ibid. 48). A directed graph belongs to non-coordinate geometry and uses G. If S contains G, the question does not arise as to which one we should choose. Furthermore, if the arithmetic of addition originates in C_{HL}-merge

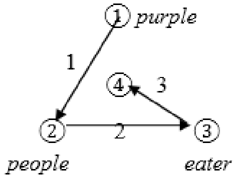
2. Possible trees and graphs for "*puple people eater*"

Consider possible structures for "purple people eater." Here, we calculate and compare the hidden balance. A graph without loops is called a tree.[2]

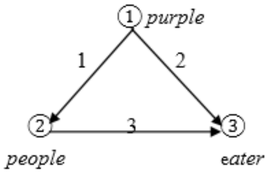
(5) String tree



(6) String tree with node 4



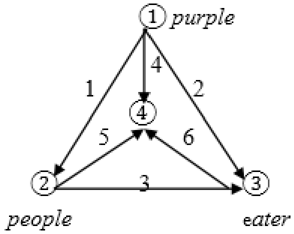
(7) Loop graph



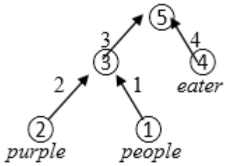
(Chomsky and McGilvray 2012: 15), the additive operation would integrate set-theoretic unordered sets {α, β} and graph-theoretic column-vector addition (linear combinations of vectors v and w: $cv + dw$) using matrix methods. Crucially, both set-theoretic and graph-theoretic additive operations respect group-theoretic axioms, in the following ways: (i) closure, i.e., the output property remains the same as the input property, (ii) the associative law, i.e., $(a \circ b) \circ c = a \circ (b \circ c)$, (iii) the existence of identity I, i.e., $a \circ I = a$, and (iv) the existence of the inverse a^{-1}, i.e., $a \circ a^{-1} = I$. Set theory and graph theory may therefore be merely two sides of the same evolution in C_{HL}-merge. Both theoretical approaches could provide different routes to the C_{HL} summit. The graph-theoretic approach could be correct both "pedagogically" and also "scientifically."

[2] A complete graph and a string tree are the two extremes with respect to the number of edges. A complete graph has the maximum number of edges $m = ½\, n\, (n - 1)$, where n is the number of nodes. A string tree has the minimum number of edges $m = n - 1$.

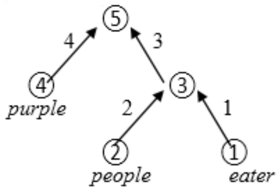
(8) Complete graph



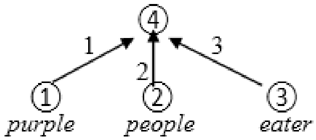
(9) Binary merge tree (a)



Binary merge tree (b)



(10) Ternary merge tree



The complete graph (8) has 16 different subgraphs (Strang 2014: 317). For example, (5), (6), (7), and (10) are subgraphs of (8). In (9), we assign smaller numbers to heads and projections from heads because they appear earlier in the structure-building space. Mathematically, numbering should be free. Linguists have accumulated empirical evidence for binary merge in (9). Is there any mathematical reason?

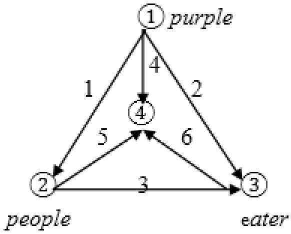
3. Theoretical background

Here, we introduce a toy model of the graph theory. The theory demonstrates how nature balances networks and distributes currents (information) to minimize loss.

3.1 Incidence matrix—Geometry

Suppose C_{HL} produces a complete graph for "purple people eater" as follows.[3]

(11) Complete graph (= (8))



The graph hides the balance. We want to know it. The balance is not "woken" yet. Later, we use a methodological "trick" to activate and reveal the balance.

We first express the graph by incidence (connectivity) matrix A. The graph is complete (i.e., all nodes are connected) and directed (i.e., all edges have directions). A expresses the geometry (topology) of the graph. The columns represent nodes bearing potential vectors (node variables: x_n) and the rows represent edges with current vectors (edge variables: y_n).[4] Assign −1 to the starting node and 1 to the end node. Tables represent matrices. Here, we omit zeros; a blank in the table indicates a zero. "The graph and

[3] We follow Strang (2008, 2009: 427-428) in introducing linear algebra, graph theory, and their application. Strang does not consider natural language examples. However, he provides a simple graph such as this that offers insight to biolinguistics.

[4] The column vectors are in the real-number vector space $\mathbf{R}^6$, and the row vectors are in $\mathbf{R}^4$.

the matrix have the same information” (Strang 2014: 313).

Table 1: Incidence matrix A of the complete graph

$A =$

	1	2	3	4
1	−1	1		
2	−1		1	
3		−1	1	
4	−1			1
5		−1		1
6			−1	1

When nodes are connected, a current (information) flows from a node with higher potential to one with lower potential. The matrix expresses potential differences (drops). Brackets are typically used to indicate that arrangements of numbers are to be interpreted as matrices. We use tables because they are easier to construct and observe.[5] The incidence matrix A expresses a system of simultaneous equations. We solve the system of simultaneous equations $\boldsymbol{Ax} = \boldsymbol{b}$. When $b = 0$, we solve $Ax = 0$, where there is no potential difference. In this example, there is no current flow. The solution x is in the nullspace of A, described as $\boldsymbol{N}(A)$, a subspace of the four-dimensional real-number vector space $\mathbf{R}^4$. This A is not solvable, i.e., infinite nonzero constant solutions (c, c, c, c) exist. The inverse A^{-1} does not exist. When $b \neq 0$, we solve $Ax = b$, where potential differences exist and the currents (information) flow. The solution x is in the column space of A, described as $\boldsymbol{C}(A)$, a subspace of $\mathbf{R}^6$. $\boldsymbol{b}$ is the potential difference e. $\boldsymbol{C}(A)$ contains solutions of $Ax = b$. Ax is the vector of differences. The components of Ax add to zero around every loop. This is Kirchhoff’s voltage law (KVL). For example, $b_1 + b_3 - b_2 = 0$, which expresses that the differences around the biggest loop consisting of edges 1, 3, –2 add to zero. Hence, $Ax = b = e$. To this point, we have described the incidence matrix A (geometry) among nodes and the potential difference.

[5] We use the Reshish matrix calculator (RMC; matrix.reshish.com) to calculate ranks.

3.2 Ohm's law—Material

Next, we consider the material (physical) property of edges. Here, Ohm's law is the key.[6] Expressed in words, Ohm's law is "*current along edge* = *conductance* times potential difference." For simplicity, assume the conductance ("how easily flow gets through" (Strang 2009: 426)) of each edge is $c = 1$, i.e., currents on all edges have equal smoothness. Hence, the conductance matrix for the entire graph is $C = I$, where I denotes the identity matrix (all zeros except 1s on the diagonal). In the same way as 1, I changes nothing in multiplication. The Ohm's law equation is $y = -Ce$, where y is a current on an edge, C is the conductance that we assume to be I, and e is the potential drop.[7] Information flows from a higher potential node (starting point) to a lower potential node (end point). We assign −1 to a starting node and 1 to the end node, resulting in a current becoming negative. Changing a current to positive requires the minus sign. Since $Ax = e$, we obtain $y = -CAx$.

3.3 Elimination (row reduction)—Cleaning a graph to get a tree

We perform Gaussian elimination (row reduction; GE) in order to reveal the essence of matrices. GE removes nonzero entries below the diagonal (pivots). Elimination changes Table 1 to Table 2. Elimination yields the echelon form (zero echelon below the diagonal) or upper triangular matrix U (nonzero entries form a triangle above including the diagonal). The shaded boxes indicate the diagonal and pivot positions. Elimination reveals the rank r of the matrix, i.e., the true size (character) of the matrix A, or "the true number of equations" (Strang 2014: 273). The rank r is the number of pivots, the number of independent rows, and the number of independent columns (ibid.). The purpose of GE is to obtain a nonzero entry in the last pivot

---

[6] If we have a spring and mass problem, we consider Hooke's law: "tension in a spring = spring constant times the elongation." This is $w_i = c_i e_i$. As in economics, the conductance c measures the cost of information flow in the C_{HL} network.

[7] "Ax versus $-Ax$ is a general headache but unavoidable" (Strang 2009: 427). Physics and electrical engineering use the minus sign whereas mechanical engineering and economics use a plus sign.

position of the lowest row in *U* of *A*. The shaded blank box (y_4, x_4) indicates that we have failed to obtain a nonzero entry and the system as it stands is unsolvable.

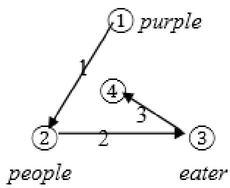
Table 2: *U* of *A*

$U(A) =$

	①	②	③	④
1	−1	1		
2		−1	1	
3			−1	1
4				
5				
6				

U shows us the essential property (“inner truth” (Strang 2009: 135)) of *A*. Infinite constant vectors (c, c, c, c) solves $Ax = 0$. We first obtain a special solution (1, 1, 1, 1) as follows. Set the free variable x_4 as 1. From $-x_3 + x_4 = 0$, we get $x_3 = 1$. From $-x_2 + x_3 = 0$, we get $x_2 = 1$. From $-x_1 + x_2 = 0$, we get $x_1 = 1$. Thus, a special solution (1, 1, 1, 1) is obtained. The complete solutions realize as a constant scalar c times (1, 1, 1, 1), which is (c, c, c, c). This is a line in $\mathbf{R}_4$. The rank (the true size of the matrix) is $r = 3$. The computation time for rank calculation is 0.472 s (the second try: 0.011 s and the third try: 0.045 s). The number of pivots counts as the rank r. *U* is expressed as a tree, i.e., a graph without loops.

(12) The true character of the complete graph: string tree with node 4



A string tree with node 4 is the identity of the complete graph. A complete graph hides maximal dependency (redundancy). MC dislikes redundancy. If MC governs C_{HL}, C_{HL} cannot have complete graphs. Elimination reduces every graph to a tree (ibid. 423). Rows are dependent when edges form a loop (ibid.). In a tree, every row (edge) is independent. In this case, we observe three independent edges y_1, y_2, and y_3. Therefore, the rank of A is $r = 3$.

3.4 Transpose matrix—Balance

Let us create the transpose of A, described as A^T, in which we exchange the rows and columns of A. "The columns of A^T are the rows of A." The entry in row i, column j of A^T comes from row j, column i of the original A: $(A^T)_{ij} = A_{ji}$ (Strang 2009: 107). Row space is expressed as $C(A^T)$, i.e., the column space of the transpose of A. The row space $C(A^T)$, i.e., the column space of transposed A, is perpendicular to $\boldsymbol{N}(A)$, which is described as $\boldsymbol{C}(A^T) \perp \boldsymbol{N}(A)$. Therefore, $\boldsymbol{C}(A^T) \cdot \boldsymbol{N}(A) = 0$. The dimension of the row space is $n - 1 = 4 - 1 = 3$, which is the rank r.

Table 3: A^T

$A^T =$

	1	2	3	4	5	6
①	−1	−1		−1		
②	1		−1		−1	
③		1	1			−1
④				1	1	1

We use A^T to calculate balance. Without external force, the equation to solve is $A^Ty = 0$, which is Kirchhoff's current law (KCL). KCL is a balance law. KCL "deserves first place among the equations of applied mathematics" (Strang 2014: 316). It expresses conservation, continuity, and balance (ibid.). We calculate how a network balances itself. The left nullspace of A^T, described as $\boldsymbol{N}(A^T)$, contains all solutions to $A^Ty = 0$. Its dimension is $m - r = 6 - 3 = 3$. The boundary conditions are as follows. y_n is a current on edge n. The minus sign indicates reverse flow.

(13) KCL boundary conditions: flow in equals flow out at each node
a. $-y_1 - y_2 - y_4 = 0$: $-y_1 - y_4 = y_2$ (Flow in equals flow out at node 1.)
b. $y_1 - y_3 - y_5 = 0$: $y_1 - y_5 = y_3$ (Flow in equals flow out at node 2.)
c. $y_2 + y_3 - y_6 = 0$: $y_2 + y_3 = y_6$ (Flow in equals flow out at node 3.)
d. $y_4 + y_5 + y_6 = 0$: $y_4 + y_5 = -y_6$ (Flow in equals flow out at node 4.)

KCL ($A^{\mathrm{T}}y = 0$) requires "no traffic jam" and "in equals out" at each node. Every loop current is a solution to KCL: $A^{\mathrm{T}}y = 0$ (Strang 2014: 313). The solutions to the KCL are: $y_1 = (1, 0, 0, -1, 1, 0)$, $y_2 = (0, 0, 1, 0, -1, 1)$, $y_3 = (0, -1, 0, 1, 0, -1)$. These three vectors are solutions to $A^{\mathrm{T}}y = 0$ and exist in $\boldsymbol{N}(A^{\mathrm{T}})$. They are independent (small) loops and are the bases of $\boldsymbol{N}(A^{\mathrm{T}})$. $y_1 + y_2 + y_3$ form the dependent (big) loop $y_4 = (1, -1, 1, 0, 0, 0)$. When the system is closed (no external force), here KCL to solve is $A^{\mathrm{T}}y = 0$. When the system is open (external force entering), KCL to solve becomes $A^{\mathrm{T}}y = f$, *where* f is the external current source. The external power source f is whatever force it requires to support the internal system balance. For a spring–mass problem, f is each mass multiplied by the gravitational constant g. Hence, $f = (m_1g, m_2g, m_3g, \dots)$ (Strang (2009: 411)).

Loops are solutions to KCL, the balance law. A tree has no loop, has no solution to KCL, and is less balanced. If we assume that C_{HL} inevitably creates loops by movement (internal merge; IM), such loops are solutions to KCL (a balance law). C_{HL} creates structures (graphs) with loops to balance the system. C_{HL} solves KCL by creating loops. IM is indispensable to a balanced structure. This may be a linear-algebraic proof that C_{HL} inevitably contains movement operation(s).

3.5 Graph Laplacian matrix $A^{\mathrm{T}}A$—Error Minimization

To observe the graph in (6) as a network and calculate its balance, we must solve a system of equations with the graph Laplacian matrix $A^{\mathrm{T}}A$ (A^{T} times A).[8] A matrix A acts on node potentials (node variables) x_{n} and we obtain

---

[8] The name *graph Laplacian matrix* comes from a discrete case of Laplacian operator Δ that appears in non-discrete (continuous) cases where differential equations expressing

the potential difference, i.e., $Ax = e$. Ohm's law $y = Ce$ governs the physical (material) property of edge currents (edge variables) y_n with conductance C. The fundamental balance equation is $A_{T}y = f$. From $Ax = e$ and $y = Ce$, we obtain $y = CAx$. From $y = CAx$ and $A^{T}y = f$, we obtain $A^{T}CAx = f$. When $C = I$, the equation is $A^{T}Ax = f$. $A^{T}A$ is the conductance matrix for the whole network.

3.5.1 $A^{T}A$ as it stands is not solvable

$A^{T}A$ is as follows.

Table 4: *Graph Laplacian matrix $A^{T}A$* (4 by 4)

$A^{T}A =$

	①	②	③	④
①	3	−1	−1	−1
②	−1	3	−1	−1
③	−1	−1	3	−1
④	−1	−1	−1	3

$A^{T}A$ is a symmetric matrix. The rule of matrix multiplication is $(AB)^{T} = B^{T}A^{T}$. Thus, $(A^{T}A)^{T} = A^{T}(A^{T})^{T} = A^{T}A$. Hence, $A^{T}A = (A^{T}A)^{T}$. The graph Laplacian is $A^{T}A = D - W$. D is the degree matrix, and the number d_{jj} indicates how many edges meet at node j. W is the adjacency matrix, and the number w_{ij} indicates whether nodes i and j are connected by an edge (Strang 2007: 148). The diagonals in D are the row sums in W. $D - W$ shows zero row sums.

$A^{T}A$ is singular. It is not invertible and not solvable (we cannot obtain a unique nonzero solution) because in addition to the zero vector (0, 0, 0, 0), infinitely many nonzero constant vectors (c, c, c, c) satisfy this system of equations. Geometrically, these infinitely many solutions form a line (subspace) in the real-number vector space $\mathbf{R}^4$. Alternatively, the left side of $A^{T}Ax = b$ adds to zero while the right side can be nonzero, which is a contradiction. Geometrically, the output is outside the line. We have no

gradient, divergence, and oscillation are used.

solution in this case. Thus, we cannot determine a unique solution and the system as it stands is not solvable. Elimination yields the U of A^TA.

Table 5: U of A^TA (3 by 4)

$U(A^TA) =$

	①	②	③	④
①	3	−1	−1	−1
②		8/3	-4/3	-4/3
③			2	-2
④				

The rank of A^TA is $r = 3$. The computation time using RMC is 0.029 s (the second try: 0.014 s). We have 0 in the last pivot position. Since the determinant det is the product of pivots, det $(A^TA) = 0$. Thus, A^TA is not invertible (solvable). A^TA has the same properties as A. If A is not invertible, A^TA is also not invertible. We cannot solve for all four potentials because the unit vector (1, 1, 1, 1) or infinitely many nonzero constant vectors (c, c, c, c) are in $\boldsymbol{N}(A)$. That is, a nonzero solution vector x satisfies $A^TAx = 0$, indicating that A^TA has no inverse. The constant vector (c, c, c, c) represents a line in $\mathbf{R}^4$. The proof is presented is the next section. The calculation of the dimension is $n - r = 4 - 3 = 1$.

3.5.2 Proof of Noninvertibility (Unsolvability) of A^TA—A fundamental fact

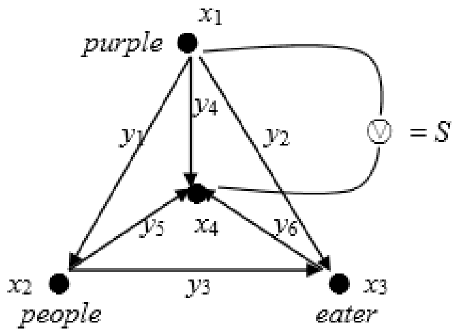
Suppose matrix A^TA is invertible. Then, the inverse matrix $(A^TA)^{-1}$ exists and $(A^TA)^{-1} \cdot A^TA = I$, where I is the identity matrix. What is the solution to $A^TAx = 0$, where x is a vector solution? Multiply both sides by the inverse. We obtain $(A^TA)^{-1} \cdot A^TAx = (A^TA)^{-1} \cdot 0$. Since $(A^TA)^{-1} \cdot A^TA = I$ and $(A^TA)^{-1} \cdot 0 = 0$, $x = 0$. The solution x must be zero. However, infinitely many nonzero solutions satisfy A^TA, i.e., infinitely many constant vectors (c, c, c, c) satisfy $A^TAu = 0$. This is a contradiction by reduction to absurdity. Therefore, the inverse matrix $(A^TA)^{-1}$ does not exist (QED).

3.5.3 Grounding—A key for making the unsolvable solvable

How can we make unsolvable $A^{\mathrm{T}}A$ solvable? $A^{\mathrm{T}}A$ can be considered as representing a system of springs and masses "lying on a table." We cannot calculate the balance, because gravitational force neutralizes the entire system, i.e., each mass and spring experiences gravity, but the masses do not move (there is no displacement) and the springs do not show tension or compression. The balance is hidden in the graph. How can we "wake up" the balance? How can we activate and calculate it? A methodological "trick" is as follows. To wake up the internal balance, we "hang the system from the ceiling" at one mass (node), so to speak. Gravity "shakes" the system, and the balance power wakes up. Gravity influences the network and pulls it down. The reaction force of the support pulls the system up, i.e., against the gravity. The system uses its balance to balance itself. Now we can calculate how the network balances itself and know what balance the system hides.

Imagine that the complete graph is a system of springs and masses, and that we hang the graph from the ceiling at node 4. This corresponds to grounding (fixing) node 4 as a support or setting the potential $x_4 = 0$ as the absolute zero temperature K. Fixing the node as a support (hanging the system at the node) makes the node potential zero, i.e., the node lacks movement or displacement. The potential $x_4 = 0$ is now known. An electrical engineer would say that node 4 is "grounded." KCL to solve becomes $A^{\mathrm{T}}y = f$, where f is the current source S coming from outside the system. "The support supplies whatever force [reaction force] it takes to support the internal forces [balance] (Strang 2008)." Reaction force (S) is whatever the support has to do to keep (fix) the displacement" (ibid). The potential becomes $x_4 = 0$; thus, S enters to compensate the network change. Imagine a balloon. Grip any area of the balloon. This corresponds to grounding the area (node) as zero. The grip (grounding a node) wakes up the hidden balance of the balloon. The grip generates the reaction force (S) that affects the entire balloon (network). KCL still holds, i.e., flow in equals flow out. S exits the grounded mass (node) and enters some other mass (node) to maintain the balance (flow in = flow out) of the entire network. In a spring and mass system, $S = (m_1g, m_2g, m_3g, m_4g)$, where m is a mass and g is the gravitational constant. In a network, $S = (x_1c, x_2c, x_3c, x_4c)$, where x is a node potential (force) and c is a certain constant expressing a reaction force required to balance the network when we fix (ground) x_4 to be zero. Assume *S* exits node 1 and enters node 4 to maintain the balance. We ground node 1 "*purple*" because it is a modifier (adjective) that is not the head of the entire projection NP. *S* enters node 4 because node 4 represents the whole structure in that it integrates the three terms.[9] We wake up and calculate the balance by "shaking" the system with *S* in this way. The hanging network is expressed as follows. *S* is an external current source. We assign potentials x_n to nodes and currents y_n to edges.

(14) The currents in a network with *S* from node 1 into node 4



The external current source connection from node 1 to node 4 does not count as a new edge. The symbol ⓥ indicates the current source *S*. What are the currents $y_1, \ldots, y_6$ on the six edges when the internal forces and *S* are in equilibrium? We calculate the potentials (voltages) x_n first, and then we calculate the currents y_n.

3.5.4 Reduced A^TA—Solving the error minimization problem

The complete graph is no longer on the desk. The network is now hanging from the ceiling at node x_1. "Gravity" (reaction force necessary for keeping the grounded-node potential to zero) changes the graph into a network that

[9] Mathematically, *S* can exit and enter any node. However, C_{HL} is a natural object and we must respect the linguistic fact that adjectives are additional information. Many problems remain.

balances itself. The external power source "shakes" the system and "wakes up" the inherent balance force. Grounding node x_1 deletes the first row and column of A^TA and gives us the reduced Laplacian matrix $A^TA_{reduced}$ as follows.

Table 6: Reduced graph Laplacian matrix $A^TA_{reduced}$ (3 by 3)

$A^TA_{reduced}$ =

	②	③	④
②	3	−1	−1
③	−1	3	−1
④	−1	−1	3

This matrix is nonsingular, invertible, and solvable, i.e., a unique nonzero solution is available. Elimination reveals the upper triangular (U; echelon form) of $A^TA_{reduced}$.

Table 7: U of $A^TA_{reduced}$

$U(A^TA_{reduced})$ =

	②	③	④
②	3	−1	−1
③		8/3	−4/3
④			2

$A^TA_{reduced}$ is square, is symmetric, has constant diagonals, and is invertible (det = 3 · 8/3 · 2 = 16). The matrix is positive definite, i.e., all pivots are nonzero and positive. The rank is $r = 3$. The computation time is 0.251 s (the second try: 0.012 s).

3.5.5 Calculation of potentials at nodes

To obtain the potentials at the nodes, we solve a matrix equation $\boldsymbol{Ux} = \boldsymbol{b}$.

(16) Potentials at the nodes

$$\underbrace{\begin{pmatrix} 3 & -1 & -1 \\ 0 & 8/3 & -4/3 \\ 0 & 0 & 2 \end{pmatrix}}_{\boldsymbol{U}} \underbrace{\begin{pmatrix} x_2 \\ x_3 \\ x_4 \end{pmatrix}}_{\boldsymbol{x}} = \underbrace{\begin{pmatrix} 0 \\ 0 \\ S \end{pmatrix}}_{\boldsymbol{b}}$$

The result of calculation of potentials at the nodes is as follows.[10]

(16) Potentials at the nodes

$x_4 = ½\, S, x_3 = ¼\, S, x_2 = ¼\, S$

The gross potential at the nodes is S. The potentials x_2 and x_3 reduce by half of x_4.

3.5.6 Calculation of currents on edges

Given the incidence matrix A, Ohm's law $y = -CAx$ yields six currents. Note $C = I$ and $x_4 = 0$. Let us reproduce the incidence matrix A.

Table 8: Incidence matrix A of the complete graph

$A =$

	①	②	③	④
1	−1	1		
2	−1		1	
3		−1	1	
4	−1			1
5		−1		1
6			−1	1

The currents on the respective edges are as follows.

---

[10] The calculation is as follows. $2x_4 = S$, $x_4 = ½\, S$; $8/3\, x_3 - 4/3\, x_4 = 0$, $8/3\, x_3 = 4/3\, x_4$, $x_3 = 3/8 \cdot 4/3\, x_4 = ½\, x_4 = ¼\, S$; $3x_2 - x_3 - x_4 = 0$, $3x_2 = x_3 + x_4$, $x_2 = 1/3\,(x_3 + x_4) = 1/3\,(¼\, S + ½\, S) = ¼\, S$.

(17) Calculation of currents on edges

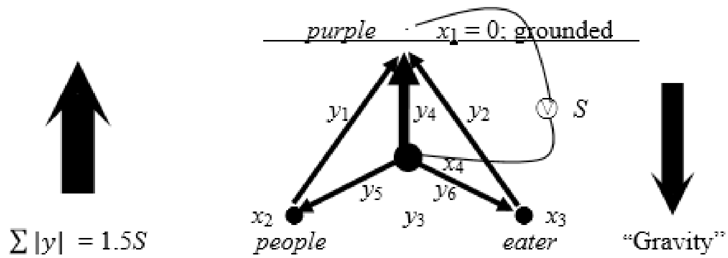
$y_1 = -(\frac{1}{4}S - 0) = -\frac{1}{4}S$
$y_2 = -(\frac{1}{4}S - 0) = -\frac{1}{4}S$
$y_3 = -(\frac{1}{4}S - \frac{1}{4}S) = 0$
$y_4 = -(\frac{1}{2}S - 0) = -\frac{1}{2}S$
$y_5 = -(\frac{1}{2}S - \frac{1}{4}S) = -\frac{1}{4}S$
$y_6 = -(\frac{1}{2}S - \frac{1}{4}S) = -\frac{1}{4}S$

The gross absolute current ($\sum |y|$) is 1.5*S*. Half of the current flows on edge 4. No current flows on edge 3. Edge 3 disappears in the balanced network. All other flows are reversed. Energy is $\frac{1}{2}S^2$.[11]

3.5.7 Balance hidden in the complete graph—Nature distributes currents to minimize information loss

In the hanging network, the balance is awake. A node-circle area indicates the potential amount, i.e., the double node area indicates the double potential. The current strength is expressed as edge thickness. The thickness corresponds to the amount that is 10 times the current: ½ *S* = 5 points, ¼ *S* = 2.5 points.

(18) Balance hidden in the complete graph



[11] To obtain energy, square each current and add them (Strang 2014: 424). The calculation is as follows. $1/16S^2 + 1/16S^2 + 1/4S^2 + 1/16S^2 + 1/16S^2 = 4/16S^2 + \frac{1}{4}S^2 = \frac{1}{2}S^2$.

This shows the "woken" balance that was hidden in the complete graph. Note that the revealed balance is symmetrical. "Gravity" means a force causing the support (node 1) to supply whatever force [reaction force] is required to support the internal forces. "Nature distributes the currents to minimize the heat loss" (Strang (2009: 428)). The potential (voltage) at node 4 is ½ *S*, which is greater than the potential at nodes 2 and 3, which are both ¼ *S*. The potential drop causes current flow in a particular direction. Recall that the potential at node 1 is grounded (fixed) as a support, i.e., the potential $x_1 = 0$. The balance works in such a way as to make half the entire current (½ *S*) exit node 1 (the fixed support) and enter node 4 (the entrance of S to maintain the balance). The rest of the four currents are equally weak (each ¼ *S*). Their direction is reversed from the original graph. Edge 3 has disappeared. That is how nature distributes the currents to minimize the heat (information) loss. Every grounded network minimizes error in an absolute sense (without comparison to others). However, the balance and preserved potentials differ among various structures (graphs). If this were a spring–mass problem, the spring (edge) 4 would be compressed with greater force (½ *S*). The other springs are compressed with weaker force (¼ *S*). Spring 3 disappears.

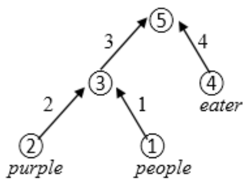
4. Balance hidden in binary merge tree and ternary merge tree

Let us calculate the balance that is hidden in the binary merge tree and ternary merge tree. These trees are familiar in linguistics literature.

4.1 Balance hidden in binary merge tree

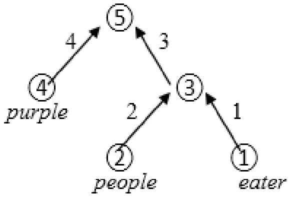
The following structure yields the "purple people" reading.

(19) Binary merge tree producing "purple people" interpretation



The following structure produces the "purple eater" reading.

(20) Binary merge tree producing "purple eater" interpretation



Both of the binary merge trees have the same incidence matrix, which expresses the topology or geometry. The nodes ①, ②, ③, ④, and ⑤ bear the potentials x_1, x_2, x_3, x_4, and x_5, respectively, and the edges 1, 2, 3, and 4 have the currents y_1, y_2, y_3, and y_4, respectively.

Table 9: Incidence matrix *A*

$A =$

	①	②	③	④	⑤
1	−1		1		
2		−1	1		
3			−1		1
4				−1	1

Unlike the incidence matrix of the complete graph, the matrix is U. The rank is $r = 4$. RMC yields the same matrix as U with a computation time of 0.232 s (the second try: 0.011 s). There are infinite nonzero constant vectors (c, c, c, c, c) that solves $Ax = 0$. The matrix is not invertible, i.e., A^{-1} does not exist. Graph theory adds constant vector solutions (c, c, c, c, c) to the vector x of potentials (Strang (2009: 423)). The product of the pivots is 1. Therefore, det = 1, which is the same as det of the identity matrix I.

Next, we consider the material property of edges. Ohm's law is the key; i.e., "*current along edge = conductance times potential difference*." Ohm's law is $y = -Ce$, where $e = Ax$. In C_{HL}, the head-projecting edges, unlike non-head-projecting edges, may have higher conductance since they constitute the backbone of phrase structures. However, for simplicity, we assume $C =$

I. That is, we treat every edge in the same way with respect to conductance. We leave a detailed analysis with various edge weights (conductance) for future research. The transpose matrix A^T is as follows.

Table 10: Transpose matrix A^T

$A^T =$

	1	2	3	4
①	−1			
②		−1		
③	1	1	−1	
④				−1
⑤			1	1

We use A^T in KCL to calculate the balance. The boundary conditions are as follows. y_n is a current on edge *n*. The minus sign indicates the reverse flow.

(21) KCL boundary conditions: flow in equals flow out at each node
  a. $-y_1 = 0$: $y_1 = 0$ (Flow in equals flow out at node 1.)
  b. $-y_2 = 0$: $y_2 = 0$ (Flow in equals flow out at node 2.)
  c. $y_1 + y_2 - y_3 = 0$: $y_1 + y_2 = y_3$ (Flow in equals flow out at node 3.)
  d. $-y_4 = 0$: $y_4 = 0$ (Flow in equals flow out at node 4.)
  e. $y_3 + y_4 = 0$: $y_4 = -y_3$ (Flow in equals flow out at node 5.)

Flow into node 1 is zero; therefore, the current on edge 1 flowing out from node 1 is zero. The same holds for nodes 2 and 4. The potentials are unknown. We do not know the potentials of x_1, x_2, and x_4. The graph Laplacian matrix A^TA is as follows.

Table 11: Graph Laplacian matrix A^TA (5 by 5)

$A^TA =$

	①	②	③	④	⑤
①	1		−1		
②		1	−1		
③	−1	−1	3		−1
④				1	−1
⑤			−1	−1	2

$A^{T}A$ is singular (unsolvable and noninvertible), because infinitely many nonzero constant vectors (*c, c, c, c, c*), a line in $\mathbf{R}^5$, exist in $\boldsymbol{N}(A)$. To make the matrix solvable, we must ground a node, thereby removing the nonzero vectors from $\boldsymbol{N}(A)$. Let us ground node 2 "purple" in (25) because it cannot be the head of the projection of the entire phrase (NP in this case). Heads develop the backbone of the phrase. We cannot ground the heads, making their potentials zero. A non-head can be grounded.

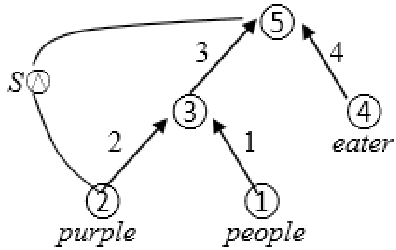
However, the assumption is controversial. In graph theory, we can ground any node. However, C_{HL} is a natural object and it is a physical fact that natural laws govern C_{HL}. We cannot say, "Anything is possible unless a contradiction arises," as in mathematics. We may not ground heads, head-projection nodes, and arguments because they build the "backbone" of phrase structures.[12]

Let us ground the adjectival node. The potential x_2 of node 2 becomes zero. Grounding a node is similar to fixing a support in a spring–mass system, where nodes = masses, and edges = springs. Before grounding, the graph is as lying on a table and the gravitational force is neutralized by the

[12] We thank the anonymous reviewer who suggested we explain and expand the island issue in this context. Islands are structural domains that prohibit extraction (Ross 1967, Huang 1982). Wh-related islands may be less controversial with regard to grounding. In graph theoretical terms, internal merge (IM) forms loops. Loops are solutions to KCL (Kirchhoff's current law), the fundamental law of equilibrium. C_{HL} uses IM to balance sentence structure. However, loops are a source of redundancy (dependence). To eliminate this redundancy, we ground the node, which is equivalent to muting a copy. In non-wh-in-situ questions of the Bulgarian or Serbo-Croatian-type, all wh-copies except the highest may be grounded, as lower copies are silent and no phonetic feature is realized in the sensorimotor system (Rudin 1988, Bošković 1997). For Chinese-type wh-in-situ questions, it is the wh-copies other than the lowest one that may be grounded, because it is the higher copies that are not pronounced. Arikawa (forthcoming) introduces a toy graph-theoretic experiment on the network-balance calculation between non-wh-in-situ *v*P (*A*) vs. wh-in-situ *v*P (*B*). Some of the findings are as follows. In terms of Gaussian elimination time, *B* is more complex (slower) than *A*, but in terms of potentials at nodes and current on edges, *A* is more complex (costly) than *B*. For the current between wh-copies created by internal merge, the flow strength is different and the direction reverses, i.e., *A* has a strong current flowing downward: higher copy → lower copy, whereas *B* has a weak current flowing upward: lower copy → higher copy. Many problems remain, however.

reaction force of the table. After grounding, the graph is hung upside down at node 2 and gravity stimulates or "shakes" the system and "wakes up" the inherent balance. Suppose that S exits node 2 and enters node 5, which is the entire structure. We want to know how the entire network balances itself with S. The following shows the network after grounding.

(22) Binary network with external current source S



How does the network balance itself? What is the potential at each node? What is the current on each edge? Grounding node 2 causes column 2 in A to disappear. Row 2 disappears in A^{T}. Column 2 and row 2 disappear in $A^{\mathrm{T}}A$. This is the reduced $A^{\mathrm{T}}A$.

Table 12: $A^{\mathrm{T}}A_{\mathrm{reduced}}$

$A^{\mathrm{T}}A_{\mathrm{reduced}} =$

	1	3	4	5
1	1	−1		
3	−1	3		−1
4			1	−1
5		−1	−1	2

$A^{\mathrm{T}}A_{\mathrm{reduced}}$ is solvable. Make $A^{\mathrm{T}}A_{\mathrm{reduced}}$ upper triangular (U) and calculate the rank.

Table 13: U of $A^{\mathrm{T}}A_{\text{reduced}}$

$U(A^{\mathrm{T}}A_{\text{reduced}}) =$

	①	③	④	⑤
①	1	−1		
③		2		−1
④			1	−1
⑤				½

In the U of $A^{\mathrm{T}}A_{\text{reduced}}$, node 2 is grounded. The rank is $r = 4$. The computation time is 0.457 s.[13] All pivots are positive. Solve the system of equations $Ux = b$ and find the potential of each node. The potentials at the nodes are as follows.[14]

(23) Potentials at the nodes
$x_5 = 2S$, $x_4 = 2S$, $x_3 = S$, $x_2 = 0$ (grounded), $x_1 = S$

The potentials add to $6S$. The network preserves a potential that is six times greater than the complete graph. Let us calculate currents. Given A, Ohm's law $y = -CAx$ yields the four currents. Assume $C = I$ and $x_2 = 0$. The calculation is as follows.

(24) Calculation of currents on edges
$y_1 = -\,(S - S) = 0$
$y_2 = -\,(S - 0) = -S$
$y_3 = -\,(2S - S) = -S$
$y_4 = -\,(2S - 2S) = 0$

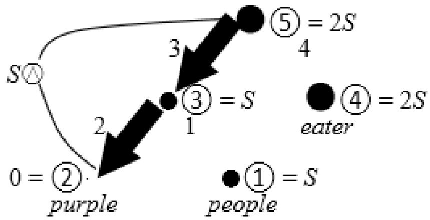
No current flows on edges 1 and 4. These edges disappear. Energy is $S^2 + S^2 = 2S^2$. Here, let us indicate how the network looks in an equilibrium state. We represent the difference of potential by node size, and the current $1S$ as

[13] The second try: 0.016 s.

[14] The calculation is as follows. $\frac{1}{2}\,x_5 = S$, $x_5 = 2S$; $x_4 - x_5 = 0$, $x_4 = x_5 = 2S$; $2x_3 - x_5 = 0$, $2x_3 = x_5$, $x_3 = \frac{1}{2} \cdot 2S = S$; $x_1 - x_3 = 0$, $x_1 = x_3 = S$.

a 10-point arrow (ten times the current); we keep the same ratio for arrows as the one adopted in Section 2.

(25) Balance hidden in the binary merge network



The gross potential at all nodes is 6*S*. The gross potential at nodes with current flow is 3*S*. The absolute gross current $\sum|y|$ is 2*S*. No current flows on edges 1 and 4, which are probe (head) projections. Those edges disappear. Given that the first merge concatenates the probe (head) and the null complement and forms a structurally symmetrical unit, we hypothesize that this is a network-theoretic expression that the complement (spell-out domain) is transferred to the external system (Footnote 19). Zero current indicates zero heat, which, in turn, indicates zero information entropy. Zero information entropy refers to zero bit, which expresses the greatest certainty, i.e., either the probability $p = 0$ (never occurs) or $p = 1$ (inevitably occurs) (Shannon 1946: 11). Assume $p = 1$, i.e., the complement information is sent to the external systems and phonetically and semantically fixed.

On the contrary, current flows on edges 2 and 3, which are specifier (modifier) projections created by the second merge. We hypothesize that this is a network-theoretic expression that specifiers are visible as goals from higher probes for further operations. These edges do not disappear. The existence of current indicates the existence of heat, which, in turn, indicates the existence of information entropy. This existence of information entropy refers to a nonzero bit. Assume that specifiers have the greatest bit (= 1). One bit expresses the greatest uncertainty, i.e., the probability is p = 0.5 (ibid). At this stage, the system is uncertain about the next step (no look-ahead). Any small piece of information is useful, and a small set of AGREE features fixes the next stage. In contrast to the upward flow on edges 2 and the tree (19) before equilibrium, the flow reverses, going downward when network equilibrium is established (25). A current flows from a higher-potential node to a lower-potential node, obeying the second law of thermodynamics (entropy law). If the upward flow is caused by "entropy," the downward (inverse) flow is caused by "negative entropy," a well-known word that originates in the sentence that we quote from Schrödinger (1944): What an organism feeds upon is negative entropy. Negative entropy (negentropy) is the source of order. The negentropy on edges 2 and 3 is the driving force of structure (order) building. The reverse flow indicates invertibility (reversibility). C_{HL} feeds upon the inverse yielding a unique solution. C_{HL} feeds upon negentropy (Footnote 18).[15]

This is a mathematical proof that "Merge is inherently asymmetrical" (Jaspers 1998, Langendoen 2003, Zwart 2009a,b, 2011, as cited in Belder and Craenenbroeck 2015: 633). C_{HL} grounds node ② to solve the system of equations, which reveals the asymmetrical balance hidden in the binary merge.[16]

---

[15] Boltzmann (Austrian physicist (statistical mechanics) and philosopher; 1844-1906) found the entropy equation $S = k.\log W$, where S is the entropy of an ideal gas, k is Boltzmann's constant (k = 3.2983×10^{-24}cal./°C), log is logarithm (the inverse of exponentiation), and W is the quantity of the gas. Schrödinger (1944) expresses the same equation as entropy = $k \log D$, where D stands for disorder. Entropy increases as D increases. Schrödinger expressed negentropy as –(entropy) = $k \log (1/D)$, where $1/D$ is the inverse of D. Negentropy (order) increases as D decreases. Given a phenomenon X and its probability $P(X)$, the equation of self-information H is $H(X) = \log_2 1/(P(X)) = -\log_2 P(X)$ (bit). Given two phenomena with the probability of one phenomena is p, the equation of information entropy is $H = -p \log_2 p$-$(1$-$p)\log_2(1$-$p)$ (bit). When $p = 0.5$ (the greatest uncertainty), $H = -\log_2 1/2 = \log_2 2 = 1$. Given M phenomena $a_1, a_2, a_3, \ldots, a_M$ with the probabilities $p_1, p_2, p_3, \ldots, p_M$, the information entropy of the source X is $H(X) = -\sum_{i=1}^{M} p_i \log_2 p_i$ (bit), which resembles the entropy equation $S = k.\log$W in natural science (Takaoka 2012: 68). More precisely, it resembles the negentropy equation.

[16] We thank the reviewer for pointing out a serious pitfall that we must avoid: that "Merge is inherently asymmetrical" does not mean that Merge is binary. In fact, a graph-theoretic calculation reveals that a ternary-merge network is also asymmetrical, as within the equilibrium, current flows in adjectival projections (the edge remains), but not in nominal projections (the edges disappear). However, a balance calculation reveals that more information is preserved (energy loss is minimized) in a binary-merge network. Thus, ternary merge is excluded by the principle of minimal computation (MC). Furthermore, given MC, Kayne's (1984) unambiguous path forces us to choose binary branching, as

It is worth noting that the two binary merge trees have distinct properties with respect to the topology of S although they share the same incidence matrix. Consider the binary merge tree where "*purple*" (node 2) and "*people*" (node 1) merge first, as in (9a). Suppose S exits node 2 and enters node 1. This is possible because node 2 "*purple*" modifies (mutually c-commands) node 1 "*people*". This modification yields the "*purple people*" reading. Voltages at nodes are as follows: $x_1 = 2S$, $x_2 = 0$, $x_3 = 2S$, $x_4 = S$, $x_5 = S$. Currents on edges are as follows: $y_1 = 0$, $y_2 = -2S$, $y_3 = S$, $x_4 = 0$. Edges 1 and 4 disappear. The flow of edge 2 reverses, whereas that of edge 3 retains the original direction. The hidden balance here is distinct from that where S exits node 2 and enters node 5, which is the entire phrase.

Consider the binary merge tree where "*people*" and "*eater*" merge first, as in (9b). Ground node 4 "*purple.*" The $U(A^{T}A_{\text{reduced}})$ resembles the identity matrix I. The rank is $r = 4$, and the computation time is 0.013 s. The hidden balance does not change when node-4-exiting S enters node 5 (the entire phrase) or node 1 (the modified term "eater"). Voltages at nodes are as follows: $x_1 = S$, $x_2 = S$, $x_3 = S$, $x_4 = 0$, $x_5 = S$. Currents on edges are as follows: $y_1 = 0$, $y_2 = 0$, $y_3 = 0$, $y_4 = -S$. All edges except edge 4 disappear. The flow of edge 4 is reversed.

What does this tell us? We hypothesize that the computation of the "*purple eater*" reading is simpler. If this is tenable, we have a linear algebraic (graph theoretic) proof that the "*purple eater*" reading is more natural: the $U(A^{T}A_{\text{reduced}})$ is more elementary and stable, i.e., the topology of S does not affect the hidden balance.

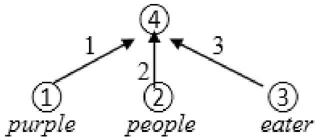
4.2 Balance hidden in ternary merge tree

Let us calculate the balance hidden in ternary merge. Assume a ternary branching tree for the same example. The tree is a subgraph of the complete graph (8, 11).

---

unary projection is too redundant, and ternary or more branching is too costly because of path ambiguity. Arikawa (2015: 106) offers a preliminary graph-theoretic analysis and evidence that supports the binary-merge network, and excludes ternary merge.

(26) Ternary merge tree



We tentatively assume that C_{HL} activates edges 1 and 2 (*purple* modifies *people*) to yield the "*purple people*" reading, while activating edges 1 and 3 (*purple* modifies *eater*) produces the "*purple eater*" reading. The incidence matrix is as follows.

Table 14: Incidence matrix A of ternary merge graph

A =

	①	②	③	④
1	−1			1
2		−1		1
3			−1	1

This matrix is already an upper triangular U. The rank is r = 3. RMC performs rank calculation to yield the same matrix as U with a computation time of 0.19 s (the second try: 0.043 s). There are infinite nonzero constant vectors (c, c, c, c, c) that solves $Ax = 0$. The matrix is not invertible, i.e., A^{-1} does not exist. Graph theory adds constant vector solutions (c, c, c, c) to the vector x of potentials. We transpose A.

Table 15: Transpose matrix A^T

A^T =

	1	2	3
①	−1		
②		−1	
③			−1
④	1	1	1

The graph Laplacian matrix A^TA is as follows.

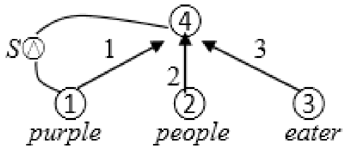
Table 16: A^TA

$A^TA =$

	①	②	③	④
①	1			−1
②		1		−1
③			1	−1
④	−1	−1	−1	3

A^TA is singular and not solvable. To solve the system, we must ground a node (Strang 2007: 155). Here, we ground node 1 "*purple*" (adjective) because it is optional (additional) information that cannot be the head of the entire noun phrase NP. Thus, the potential at node 1 becomes zero. Therefore, $x_1 = 0$. Assume that the external power source exits node 1 and enters node 4 (the entire phrase).

(27) Ternary merge network with external current source S



How does the network balance? What is the potential at each node when the system is in equilibrium? What is the current on each node? Grounding node 1 causes column 1 in A to disappear. Row 1 also disappears in A^T. Column 1 and row 1 disappear in A^TA, which becomes a reduced A^TA.

Table 17: A^TA_{reduced}

$A^TA_{\text{reduced}} =$

	②	③	④
②	1		−1
③		1	−1
④	−1	−1	3

$A^{\mathrm{T}}A_{\text{reduced}}$ is no longer singular. It is invertible. We perform GE to make $A^{\mathrm{T}}A_{\text{reduced}}$ upper triangular (U) and calculate the rank.

Table 18: U of $A^{\mathrm{T}}A_{\text{reduced}}$

$U(A^{\mathrm{T}}A_{\text{reduced}})$ =

	②	③	④
②	1		−1
③		1	−1
④			1

Node 1 is grounded. The rank is $r = 3$, all pivots are positive, and Det = 1. The computation time is 0.418 s (the second try: 0.411 s, the third try: 0.418 s, the fourth try: 0.012 s). Solve the system of equations and find the potential of each node.[17]

(28) Potentials at the nodes
$x_4 = S, x_3 = S, x_2 = S, x_1 = 0$ (grounded)

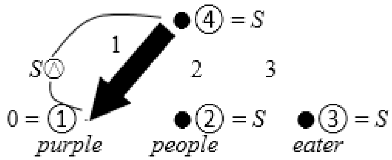
The node potentials add to $3S$, which is half that of the binary network. Less force is preserved in the ternary merge. In other words, information loss is not minimized in the ternary merge. Given the incidence matrix A, Ohm's law $y = -CAx$ yields the four currents. Assume $C = I$ and $x_1 = 0$. The calculation is as follows.

(29) Calculation of currents on the edges
$y_1 = -(S - 0) = -S$
$y_2 = -(S - S) = 0$
$y_3 = -(S - S) = 0$

There is no current on edges 2 and 3 (no potential drop); thus, these edges disappear. Energy is S^2. Here, we show how the system appears in an equilibrium state.

[17] The calculation is as follows. $x_4 = S$; $x_3 - x_4 = 0$, $x_3 = x_4 = S$; $x_2 - x_4 = 0$, $x_2 = x_4 = S$.

(30) Balance hidden in the ternary merge network



The gross potential at all nodes in the ternary merge is 3*S*, which is half that of binary merge. The gross potential at nodes with current flow is *S*, which is a third of that of binary merge. The gross current on the existing edge is *S*, which is half that of binary merge. No current flows on edges 2 and 3 because no potential drop exists. Hence, edges 2 and 3 disappear. If this is interpreted as a spring–mass problem, spring 1 is compressed by force *S*. The force *S* pulls the ternary merge network. Recall that the force 2*S* pulls a binary merge network. More energy is preserved in the binary merge network. Energy loss is minimized there.

5. Comparison

The following table summarizes the network-theoretic differences among the trees and graphs. We consider matrix properties as follows. (a) the number of nodes, (b) the number of edges, (c) the number of loops, (d) the rank calculation and the computation time of *A*, (e) the number of dependent edges (edge minus rank), (f) the rank calculation and the computation time of $A^{T}A_{\text{reduced}}$, (g) the gross potential at all nodes after grounding, (h) the gross potential at nodes with current after grounding, (i) the absolute current (∑|y|) after grounding, (j) the number of vanished edge after grounding, and (k) the energy in the entire system. See appendix for balance calculations for a string tree, a string tree with node 4, and a loop graph.

What makes binary merge special? The rank is $r = 4$, which is the greatest and the matrix size is the largest. A binary merge network contains the greatest amount of information. With respect to the gross potential at nodes after grounding, the binary merge and the string tree with node 4 preserve the greatest potential: 6*S*. The conservation law appears to favor

Table 17: Comparison of matrix property and balance of trees and graphs

	Node	Edge	Loop	A		Dependent edge	$A^T A_{\text{reduced}}$		After grounding				
				Rank	Time (s)		Rank	Time (s)	Gross potential at all nodes	Potential at node with current	Absolute current $\sum\lvert y\rvert$	Edge vanished	Energy
Complete graph	4	6	3	3	0.472 (0.011)	3	3	0.251 (0.012)	S	S	$3/2S$ = $1.5S$	1	$1/2S^2$
Binary tree	5	4	0	4	0.232 (0.011)	0	4	0.457 (0.016)	$6S$	$3S$	$2S$	2	$2S^2$
Ternary tree	4	3	0	3	0.19 (0.043)	0	3	0.418 (0.012)	$3S$	S	S	2	S^2
String tree	3	2	0	2	0.07	0	2	0.039	$3S$	$3S$	$2S$	0	$2S^2$
String tree with node 4	4	3	0	3	0.077	0	3	0.066	$6S$	$6S$	$3S$	0	$3S^2$
Loop graph	3	3	1	2	0.015	1	2	0.013	S	S	$4/3S \approx 1.3S$	0	$2/3S^2$

both networks. What is the difference between the two networks? The grounded binary merge network with the revealed balance is significantly different from the original tree; two edges disappear in the balanced binary merge. Unlike the string tree with node 4, which shows the optimal balance from the beginning, the binary merge network lacks optimal balance. Note that string trees as well as the loop graph are balanced from the beginning, i.e., no edge vanishes. The string trees are too balanced for C_{HL} to adopt. C_{HL} prefers inherent unbalance.[18] To summarize, C_{HL} chooses a binary merge network for two reasons. First, the binary merge can contain more information. Second, the binary merge lacks equilibrium, (i.e., it is unbalanced). The binary merge hides the balance, as shown by the number of vanished edges after grounding. C_{HL} chooses the binary merge because of its large capacity and unbalance.

---

[18] We thank the reviewer for pointing out that the relevance of the entropy and inertia observed in C_{HL} is being investigated by Massimo Piattelli-Palmarini and Juan Uriagereka. The greatest information entropy (1 bit, i.e., a state in which any small piece of information is most useful) corresponds to the greatest informational symmetry (p = 0.5, i.e., everything is uncertain and every future looks similar) (Shannon 1948: 11). "Organic evolution has its physical analogue in the universal law that the world tends, in all its parts and particles, to pass from certain less probable [order] to more probable configurations or states [disorder]. This is the second law of thermodynamics. It has been called *the law of evolution of the world*; and we call it, after Clausius, the Principle of *Entropy*, which is a literal translation of *Evolution* into Greek" (Thompson 1942: 11). Entropy "is not a hazy concept or idea, but a measurable physical quantity … At the absolute zero point of temperature (roughly −273°C) the entropy of any substance is zero. … [T]he unit in which entropy is measured is cal./°C (just as the calorie is the unit of heat or the centimeter the unit of length.)" (Schrödinger 1944). Uriagereka (2000: 79) hints that "dissipative phase transitions, corresponding to **symmetry breaking of equilibrium states**, are behind the origin of form in life." Piattelli-Palmarini and Uriagereka (2008: 219-220) hypothesize that "the elegant syntactic molecule," in which a probe (head) and the complement undergo first merge and specifiers undergo additional merges on "derivational buffers" in their "derivational dimension," is created by "two opposing forces": the "repulsion" and "gluing" forces. The gluing force works to transfer the complement to the external systems for equilibrium, while the repulsion force works to grow the structure, breaking symmetry. In our hierarchical binary network in equilibrium, no current flows

A complete graph contains the greatest dependency (redundancy), i.e., it has three dependent edges. If the minimal computation principle (MC) governs C_{HL}, C_{HL} cannot adopt the complete graph because it contains too much redundancy. On the other hand, the complete graph contains the greatest number of loops. Loops are solutions to KCL, i.e., the balance law. The complete graph has the greatest balance; information flows around the loops and the network is stable and in equilibrium. C_{HL} does not adopt this highly balanced network as the building block because it is too symmetrical.

The computation time (the first attempt) for GE of A is the longest for the complete graph. The computation time (the first attempt) for GE of $A^{T}A_{\text{reduced}}$ is the longest for the binary merge network. The computation time of GE is not necessarily faster for binary merge. However, the computation time we obtain from RMC might contain coincidence (Andrew Shishkin (p.c.)). The string trees are one-dimensional lines. The complete graph and the loop graph are two-dimensional planes that contain loops. The ternary merge tree has one level hierarchically. The binary merge tree has two levels hierarchically. Two heads (levels) are better than one, so to speak.

6. Conclusion

String trees are balanced from the beginning. They do not hide the balance but are balanced in the first place. The binary merge lacks redundancy, i.e., it lacks loops. The binary merge is not balanced and it hides the balance to the greatest degree. C_{HL} requires asymmetry (unbalance) to initiate structure building. The current in the binary merge network is not minimized, i.e., it contains error (loss). This does not mean that C_{HL} disobeys MC. Because the binary merge network contains errors, there is room for MC to work in the

---

in the probe-complement projections, whereas current flows in the specifier projections. This provides a network-theoretic proof of the two opposing forces, i.e., the repulsion force persists and continues to grow the structure and break symmetry for the next phase, while the gluing force disappears in equilibrium after the complement has been transferred to the external systems (sensorimotor system and conceptual-intentional system). The reviewer's comment is significant in that it suggests a connection between Piattelli-Palmarini and Uriagereka's (2008) project and our network-theoretic analysis.

computation. MC may be a member of natural laws, such as the principle of least action, inertia law (Newton's first law), and the conservation law. Nature has selected the binary merge not because the error is zero but because the binary merge can contain more information and the error is not minimized. This asymmetrical (unbalanced) state hides the driving force for infinite structural growth of a binary branching tree. We have graph-theoretical and physical reasons for concluding that the syntactic operation *merge* must be binary.

C_{HL} obeys MC (Chomsky 1995, 2005). Study of any growing system, including C_{HL}, considers three factors (ibid.). The first factor is environment (input data), the second genetics, and the third natural laws. MC is the third factor. Chomsky attempts to reduce the C_{HL} phenotype to the third factor. If possible, biolinguistics will be integrated into physics. Both fields must undergo breakthroughs in the future.

Acknowledgements

We thank the two anonymous reviewers for their insightful comments and suggestions. Without their advice, this paper would probably not have been of great interest even to the "quite limited audience for this type of article," still less to those "linguists who would not be swayed by any of these arguments, as they will prefer empirical evidence." We are grateful to editors of the journal for their decision that the paper can make a contribution to the field. I would like to express my gratitude to Massimo Piattelli-Palmarini, who welcomed me to join a fascinating boilinguistics project at a stimulating MIT lecture in 2003, and to Lyle Jenkins, for his insightful lecture on future Galois-theoretic biolinguistics in Massimo's class. Finally, we would like to thank Enago (www.enago.jp) for the English language review.

References

Arikawa, K. 2013. Is word order asymmetry mathematically expressible? *Biolinguistics* 7: 276-300.

Arikawa, K. 2015. Graph-theoretic and physical account of binary versus ternary branching structure in human natural language computation. *St. Andrew's University Bulletin of the Research Institute.* Vol. 41, No. 2, 79-108. Osaka, Japan.

Arikawa, K. Forthcoming. Does Gaussian elimination teach about uninterpretable feature elimination in C_{HL}? *St. Andrew's University Bulletin of the Research Institute*. Vol. 42, No. 3. Osaka, Japan.

Belder, M. D. and J. V. Craenenbroeck. 2015. How to merge a root. *Linguistic Inquiry* 46: 625-655.

Bošković, Ž. 1997. Multiple wh-fronting and economy of derivation. *West Coast Conference on Formal Linguistics* 16: 49-63.

Chomsky, N. 1957. *Syntactic structures.* The Hague: Mouton.

Chomsky, N. 1995. *The Minimalist Program*. The MIT Press.

Chomsky, N. 2005. Three factors in language design. *Linguistic Inquiry* 36: 1-22.

Chomsky, N. and J. McGilvray. 2012. *The science of language: interviews with James* McGilvray. Cambridge: Cambridge University Press.

Fukui, N. 1996. On the nature of economy in language. *Cognitive Studies* 3, 51-71.

Hale, K. 1982. Preliminary remarks on configurationality. *North East Linguistic Society* 12, 86-96.

Honma, T. 1975. *Gurahu Riron Nyuumon* [Graph theory: An introduction]. Koodansya.

Huang, C.-T. J. 1982. Logical relations in Chinese and the theory of grammar. Doctoral dissertation, MIT.

Jaspers, D. 1998. Categories and recursion. *Journal of Applied Linguistics* 12: 81-112.

Kayne, R. 1984. *Connectedness and binary branching.* Dordrecht, Foris.

Kayne, R. 1994. *The antisymmetry of syntax.* Cambridge, MA: MIT Press.

Langendoen, D. T. 2003. Merge. In Carnie, A, H. Harley, and Willie, M-A (eds) *Formal approaches to function in grammar: In honor of Eloise Jelinek*, 307-318. Amsterdam: John Benjamins.

Lasnik, H. 2000. *Syntactic structures revisited: contemporary lectures on classic transformational theory*. Cambridge, MA: MIT Press.

Miyagawa, S. 1989. *Structure and Case Marking in Japanese*. San Diego, Calif.: Academic Press.

Newton, I. 1687. *Philosophiæ Naturalis Principia Mathematica* [Mathematical principles of natural philosophy].

Piattelli-Parmarini, M. and J. Uriagereka. 2008. Still a bridge too far? Biolinguistic questions for grounding language on brains. *Physics of Life Reviews* 5, 207-224.

Ross, J. R. 1967. Constraints on variables in syntax. Doctoral dissertation, MIT.

Published as *Infinite syntax*! Norwood, N.J.: Ablex (1986).

Rudin, C. 1988. On multiple questions and multiple wh-fronting. *Natural Language and Linguistic Theory* 6: 445-501.

Saito, M. and Hoji, H. 1983. Weak crossover and move α in Japanese. *Natural Languages and Linguistic Theory* 1, 245-259.

Shannon, C. E. 1948. A mathematical theory of communication. *The Bell System Technical Journal*, Vol. 27, 379-423, 623-656.

Schrödinger, E. 1944. *What Is Life? The Physical Aspect of the Living Cell.* Cambridge: Cambridge University Press (1967).

Stewart, I. 1995. *Concepts of Modern Mathematics*. New York: Dover.

Strang, G. 2007. *Computational Science and Engineering*. Wellesley-Cambridge Press.

Strang, G. 2008. Computational science and engineering videos. Lecture 1 (Four special matrices), 12 (Graphs and networks) and 13 (Kirchhoff's current law). MITOCW.

Strang, G. 2009. *Introduction to Linear Algebra (4th edition)*. Wellesley-Cambridge Press.

Strang, G. 2014. *Differential Equations and Linear Algebra*. Wellesley-Cambridge Press.

Strang, G. 2015. 18.06 Linear Algebra videos. Lecture 2 (Elimination with matrices) and 15 (Projections onto subspaces). MITOCW.

Takaoka, E. 2012. *Shanon-no Jyoohoo Riron Nyuumon* [Shannon's information theory, an introduction]. Tokyo: Koodansya.

Thompson, D'Arcy W. 1942. *On Growth and Form*. (The complete revised edition, 1992, New York: Dover.)

Uriagereka, J. 2000. *Rhyme and reason: an introduction to minimalist syntax*. Cambridge: MIT Press.

Zwart, C. J-W. 2009a. Prospects for top-down derivation. *Catalan Journal of Linguistics* 8: 161-187.

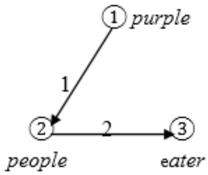
Zwart, C. J-W. 2009b. Uncharted territory? Towards a non-cartographic account of Germanic syntax. In Alexiadou, A, J. Hankamer, T. McFadden, J. Nuger, and F. Schäfer (eds) *Advances in comparative Germanic syntax*, 59-84. Amsterdam: Benjamins.

Zwart, C. J-W. 2011. Structure and order: Asymmetric Merge. In Boeckx, C. (ed) *Oxford handbook of linguistic Minimalism*, 96-118. Oxford: Oxford University Press.

Appendix

1. Balance hidden in a string tree

(1) String tree[19]



The incidence matrix is as follows.

Table 1: Incidence matrix *A*

$A =$

	1	2	3
1	−1	1	
2		−1	1

The *U* of *A* is as follows.

Table 2: *U* of *A*[20]

$U(A) =$

	1	2	3
1	−1	1	
2		−1	1

[19] Every string tree starts from node zero (Strang 2009: 22). We omit node zero. The linguistic translation is as follows. Every V starts from node zero. “When an element {α} is the first one to be taken from the resource by Unary Merge, it is included into an empty derivational workspace; that is, the object under construction is the empty set ∅ (see also Zwart 2011: 102)” (Belder and Craenenbroeck 2015: 636). If Unary Merge merges zero node and V, the linear corresponded axiom (LCA; Kayne 1994) predicts that the most unmarked order is <Obj, V>, contrary to Kayne (1994)’s conclusion that the most unmarked order is <V, Obj>. See Arikawa (2013: 283) for a similar analysis proposed independently.

[20] The rank is $r = 2$. The computation time is 0.07 s. *U* is expressed as the same

The transpose of A is as follows.

Table 3: A^{T}

$A^{\mathrm{T}} =$

	1	2
①	−1	
②	1	−1
③		1

The graph Laplacian matrix $A^{\mathrm{T}}A$ is as follows.

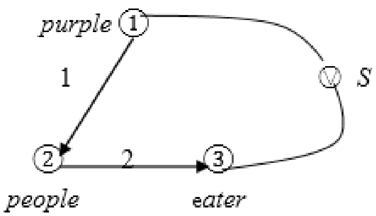
Table 4: Graph Laplacian matrix $A^{\mathrm{T}}A$

$A^{\mathrm{T}}A =$

	①	②	③
①	1	−1	0
②	−1	2	−1
③	0	−1	1

The grounded network is as follows.[21]

(2) A grounded string tree with the external current source



string tree as the original string tree. There are infinite nonzero constant vectors (*c, c, c, c, c*) that solves $Ax = 0$. The matrix is not invertible, i.e., A^{-1} does not exist. Graph theory adds constant vector solutions (*c, c, c*) to the vector x of potentials. The string tree shows the true character as it is.

[21] $A^{\mathrm{T}}A$ is not solvable. To make $A^{\mathrm{T}}A$ solvable, we ground node 1. The potential is $x_1 = 0$. As the reaction force of grounding, the external current source exits node 1 and enters node 3.

The reduced graph Laplacian, where row 1 and column 1 are deleted, is as follows.

Table 5: Reduced graph Laplacian matrix $A^{T}A_{reduced}$

$A^{T}A_{\text{reduced}} =$

	②	③
②	2	−1
③	−1	1

GE reveals U of $A^{T}A_{reduced}$.

Table 6: U of $A^{T}A_{reduced}$[22]

$U(A^{T}A_{\text{reduced}}) =$

	②	③
②	2	−1
③		½

The node potentials are as follows.[23]

(3) Potentials at the nodes
$x_3 = 2S$, $x_2 = S$, $x_1 = 0$ (grounded)

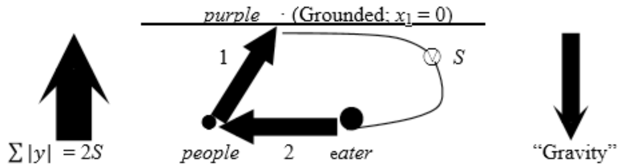
The gross potential at the nodes is $3S$. Given A, Ohm's law $y = -CAx$ yields six currents. $C = I$ and $x_1 = 0$. The currents on the respective edges are as follows.

(4) Calculation of currents on edges
$y_1 = -(S - 0) = -S$, $y_2 = -(2S - S) = -S$

The gross absolute current ($\sum|y|$) is $2S$. The two edges show the reversed flow with the same current strength. Energy is $2S^2$. Let us visualize the balance hidden in the ternary branching tree.

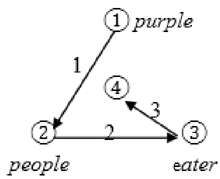
[22] The rank is $r = 2$. The computation time is 0.039 s.
[23] The calculation is as follows. $\frac{1}{2}x_3 = S$, $x_3 = 2S$; $2x_2 - x_3 = 0$, $2x_2 = 2S$, $x_2 = S$.

(5) Balance hidden in the string tree



2. Balance hidden in a string with node 4

(6) String tree with node 4



The incidence matrix is as follows.

Table 7: Incidence matrix *A*

$A =$

	①	②	③	④
1	−1	1		
2		−1	1	
3			−1	1

The *U* of *A* is as follows.

Table 8: *U* of *A*[25]

$U(A) =$

	①	②	③	④
1	−1	1		
2		−1	1	
3			−1	1

---

24 A string tree with node 4 is the identity of the four-node-complete graph.

25 This is the same as the incidence matrix *A*. The string tree shows the identity as it is. The rank is $r = 3$. The computation time is 0.077 s. There are infinite nonzero constant vectors (*c, c, c, c, c*) that solves $Ax = 0$. The matrix is not invertible, i.e., A^{-1} does not exist. Graph theory adds constant vector solutions (*c, c, c, c*) to the vector *x* of potentials.

The transpose of A is as follows.

Table 9: A^{T}

$A^{\mathrm{T}} =$

	1	2	3
①	−1		
②	1	−1	
③		1	−1
④			1

The graph Laplacian matrix $A^{\mathrm{T}}A$ is as follows.

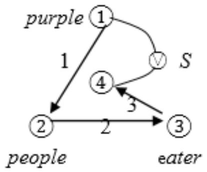
Table 10: Graph Laplacian matrix $A^{\mathrm{T}}A$

$A^{\mathrm{T}}A =$

	①	②	③	④
①	1	−1		
②	−1	2	−1	
③		−1	2	−1
④			−1	1

The grounded network is as follows.[26]

(7) The string tree with node 4 with the external current source S



The reduced graph Laplacian, where row 1 and column 1 are deleted, is as follows.

Table 11: Reduced graph Laplacian matrix $A^{\mathrm{T}}A_{reduced}$

$A^{\mathrm{T}}A_{\mathrm{reduced}} =$

	②	③	④
②	2	−1	
③	−1	2	−1
④		−1	1

[26] $A^{\mathrm{T}}A$ is not solvable. To make $A^{\mathrm{T}}A$ solvable, we ground node 1. The potential becomes $x_1 = 0$. Assume S exits node 1 and enters node 4.

Elimination reveals U of $A^{\mathrm{T}}A_{reduced}$.

Table 12: U of $A^{\mathrm{T}}A_{reduced}$

$A^{\mathrm{T}} =$

	1	2	3
①	−1		
②	1	−1	
③		1	−1
④			1

$r = 3$. The computation time is 0.066 s. The node potentials are as follows.[27]

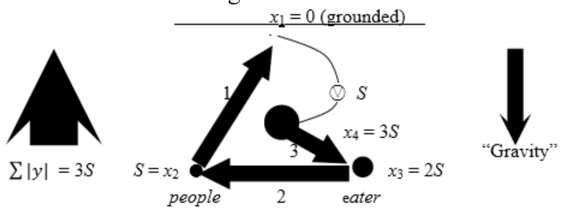
(8) Potentials at the nodes
$x_4 = 3S, x_3 = 2S, x_2 = S, x_1 = 0$

The gross potential at the nodes is $6S$. Given A, Ohm's law $y = -CAx$ yields six currents. Note $C = I$ and $x_1 = 0$. The currents on the respective edges are as follows.

(9) Calculation of currents on edges
$y_1 = -(S - 0) = -S, y_2 = -(2S - S) = -S, y_3 = -(3S - 2S) = -S$

The gross absolute current is $3S$. Flows on all edges are reversed. Energy is $3S^2$.

(10) Balance hidden in the string tree with node 4[28]

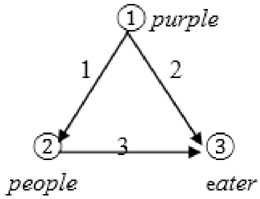


[27] The calculation is as follows. $1/3x_4 = S$, $x_4 = 3S$; $3/2x_3 - x_4 = 0$, $3/2x_3 = x_4$, $x_3 = 2/3 \cdot 3S = 2S$; $2x_2 - x_3 = 0$, $2x_2 = 2S$, $x_2 = S$.

[28] The string trees have no loops. They hide the invisible starting point $x_0 = 0$, which adds a dimension. Geometrically, the two-edge string tree is expressed as the linear combination filling the entire three-dimensional space. The three-edge tree is expressed as the linear combination of the entire four-dimensional space. A unique solution solves the system. We can pinpoint the unique solution in the entire space. Thus, the string trees are invertible and nonsingular (Strang 2009: 22-27).

3. Balance hidden in a loop graph

(11) Loop graph[29]



The incidence matrix is as follows.

Table 13: Incidence matrix *A*

$A =$

	1	2	3
1	−1	1	
2	−1		1
3		−1	1

The *U* of *A* is as follows.

Table 14: *U* of *A*

$U(A) =$

	1	2	3
1	−1	1	
2		−1	1
3			

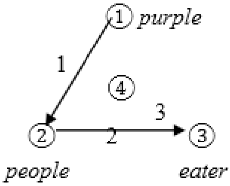
U is expressed as a tree without edge 3.

---

[29] The loop graph contains cyclic differences. Strang (2009: 25) shows a similar example. The arrow direction shows that the adjective *purple* modifies either *people* or *eater*.

[30] The rank is $r = 2$. The computation time is 0.015 s. The matrix is unsolvable, i.e., A^{-1} does not exist.

(12) The true character of the loop graph



The true identity of the loop graph is the string tree. The transpose of A is as follows.

Table 15: A^{T}

$A^{\mathrm{T}} =$

	1	2	3
①	−1	−1	
②	1		−1
③		1	1

The graph Laplacian matrix $A^{\mathrm{T}}A$ is as follows.

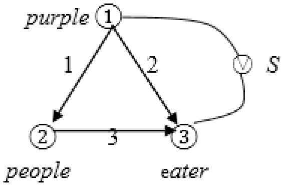
Table 16: Graph Laplacian matrix $A^{\mathrm{T}}A$

$A^{\mathrm{T}}A =$

	①	②	③
①	2	−1	−1
②	−1	2	−1
③	−1	−1	2

The grounded network is as follows.[31]

(13) The loop graph with the external current source S



[31] $A^{\mathrm{T}}A$ is unsolvable. To make $A^{\mathrm{T}}A$ solvable, let us ground node 1. The potential of node 1 is $x_1 = 0$.

The reduced graph Laplacian matrix is as follows.

Table 17: Reduced graph Laplacian matrix $A^{\mathrm{T}}A_{reduced}$

$A^{\mathrm{T}}A_{\mathrm{reduced}} =$

	2	3
2	2	−1
3	−1	2

Elimination reveals U of $A^{\mathrm{T}}A_{reduced}$.

Table 18: U of $A^{\mathrm{T}}A_{reduced}$ [32]

$U(A^{\mathrm{T}}A_{\mathrm{reduced}}) =$

	2	3
2	2	−1
3		3/2

The node potentials are as follows.[33]

(14) Potentials at the nodes
$x_3 = 2/3S$, $x_2 = 1/3S$, $x_1 = 0$ (grounded)

The gross potential at the nodes is S. Given A, Ohm's law $y = -CAx$ yields six currents. Note $C = I$ and $x_1 = 0$. The currents on the respective edges are as follows.

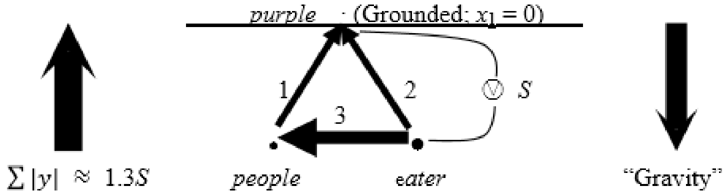
(15) Calculation of currents on edges
$y_1 = -(1/3S - 0) = -1/3S$, $y_2 = -(2/3S - 0) = -2/3S$,
$y_3 = -(2/3S - 1/3S) = -1/3S$

The gross absolute current is $4/3S$. Flows on all edges are reversed. The flow on edge 2 is the strongest. Energy is $2/3S^2$.[34] Let us visualize the balance in the loop graph.

---

32 The rank is $r = 2$. The computation time using RMC is 0.013 s.
33 The calculation is as follows. $3/2x_3 = S$, $x_3 = 2/3S$; $2x_2 - x_3 = 0$, $2x_2 = 2/3S$, $x_2 = 1/3S$.
34 The calculation is as follows. $(-1/3S)^2 + (-2/3S)^2 + (-1/3S)^2 = 1/9S^2 + 4/9S^2 + 1/9S^2 = 2/3S^2$.

(16) Balance hidden in the loop graph[35]



[35] The loop graph has a loop, and the system has cyclic differences. Geometrically, the three-edge loop graph is expressed as the linear combination filling only the subspace (two-dimensional plane $b_1 + b_2 + b_3 = 0$) in the entire three-dimensional space. The system has infinitely many solutions on the plane or the system has no solution if the linear combination is outside the plane. Thus, we cannot obtain a unique solution. We cannot pinpoint the unique solution in the entire space. Therefore, the loop graph is noninvertible and singular (Strang 2009: 22-27).